SCIENCE, TECHNOLOGY, ENGINEERING, AND MATHEMATICS CAREER CLUSTER ENDORSEMENT

BHSU & SDSMT

Endorsement coursework requirements must include courses from each Strand totaling 15 or more credits

Strand 1	**Robotics (3 or more credits)**
Strand 2	**Engineering (3 or more credits)**
Strand 3	**Nine credits in one or more of the following**
Option 1	*Aviation*
Option 2	*Computer science/programming*
Option 3	*Energy*
Option 4	*Electronics*

Courses Meeting the Requirement	CSC 415 (3 Credits)	CENG 244/L (3 Credits)	CSC 340 (3 Credits)	CSC 215 (4 Credits)	CSC 315 (4 Credits)	CSC 150 (3 Credits)	CSC 170/L (3 Credits)	CSC 210 (1 Credits)
Strand 1	X							
Strand 2		X	X					
Strand 3 *(Option 1)*								
Strand 3 *(Option 2)*				X	X	X	X	X
Strand 3 *(Option 3)*								
Strand 3 *(Option 4)*								

Total Required Credits through University: 15

Course Number	Course Name	University Course Description	Method	Sessions Offered	Total Credits
CSC 415	Introduction to Robotics	An introduction to mechatronic systems and embedded systems for robotics. This course will cover the basics required for autonomous mobile robotics. The course will begin with a survey of existing systems and some background mathematics. Core course topics will include electromechanical components, electronics for motor control, sensors and instrumentation, mobile robotic kinematics and movement, microcontrollers, real time computing, and embedded system design and development. Course projects will include student teams building task oriented mobile robots with emphasis on the hardware development.	In-person	Odd Fall	3
CENG 244/L	Introduction to Digital Systems/Lab	This course is designed to provide students with an understanding of the basic concepts of digital systems and their hardware implementation. Topics covered include combinational logic circuits, sequential logic circuits, and CPU control.	In-person	Fall & Spring	3
CSC 340	Software Engineering and Design	An introduction to the software engineering process including a survey of development methodologies (waterfall, iterative, incremental, agile). The class includes modules on fundamental software engineering tools and skills in the areas of testing, test plan development, performance analysis and tuning, and requirements analysis. Teams and teaming are a central theme supported by extensive use of project management systems for communication, source code/revision control, and project estimation.	In-person	Fall & Spring	3
CSC 150	Computer Science I	An introduction to computer programming. Focus on problem solving, algorithm development, design, and programming concepts. Topics include sequence, selection, repetition, functions, and arrays.	In-person	Fall & Spring	3
CSC 170/L	Programming for Engineers and Scientists	This course teaches engineering and science students how to write software to solve problems in their disciplines. The emphasis on mathematics, numerical methods, and development for embedded systems will prepare students to incorporate computer solutions into courses in their majors. Topics will include an overview of the software development process, development environments, algorithm development, control structures, internal and external storage, testing, and debugging. Additional topics may include programming robots and processing sensor data.	In-person	Fall & Spring	3

Course Number	Course Name	University Course Description	Method	Sessions Offered	Total Credits
CSC 210	C++ Introduction for Programmers	Learn to transition from Java, Python or other programming languages to C++. Explore the differences in memory management, classes, and pointers to become an effective C++ programmer. Learn to work with the Standard Template Library to create concise, efficient, and readable programs in C++.	In-person	Spring	1
CSC 215	Programming Techniques	This is a course that builds on the concepts and techniques introduced in Computer Science 1. Topics include binary files, bit manipulation, memory management, recursion, linked lists, stacks, queues and object oriented programming. Problem solving, algorithm design, standards of program style, debugging and testing are emphasized in this course.	In-person	Fall & Spring	4
CSC 315	Data Structures & Algorithms	A systematic study of data structures and accompanying algorithms with an emphasis on implementation and algorithmic complexity. Program development is done on Linux systems using standard software engineering tools. Topics may include: principles of object-oriented programming, such as inheritance, abstraction, polymorphism, encapsulation, and late binding; binary and m-ary trees, heaps, maps, sets, disjoint sets, and graphs; sorting techniques, hashing, shortest path and minimal spanning tree algorithms, string matching algorithms and an introduction to dynamic programming.	In-person	Fall & Spring	4